IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Patent Application

Applicant(s): A.Q. Khan et al.
Case: 4-14-28
Serial No.: 10/722,933
Filing Date: November 26, 2003
Group: 2619
Examiner: Brian T. O'Connor

Title: Processor with Scheduler Architecture Supporting Multiple Distinct Scheduling Algorithms

---

APPEAL BRIEF

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

Applicants (hereinafter "Appellants") hereby appeal the final rejection dated February 5, 2008 of claims 1-20 of the above-identified application.

REAL PARTY IN INTEREST

The present application is assigned of record to Agere Systems Inc. On April 2, 2007, the assignee Agere Systems Inc. completed a merger with LSI Logic Corporation, with the resulting entity being named LSI Corporation. LSI Corporation is the real party in interest.

RELATED APPEALS AND INTERFERENCES

As noted in the present specification at page 1, lines 5-13, the present invention is related to the invention described in common-assigned and concurrently-filed U.S. Patent Application Serial No. 10/085,219 entitled "Processor With Dynamic Table-Based Scheduling Using Linked Transmission Elements For Handling Transmission Request Collisions." This application is the subject of a pending Board appeal.

STATUS OF CLAIMS

The present application was filed on November 26, 2003 with claims 1-20. Claims 1-20 are currently pending in the application. Claims 1, 19 and 20 are the independent claims.

Claims 1-20 stand finally rejected. Claims 1-20 are appealed.

STATUS OF AMENDMENTS

There have been no amendments filed subsequent to the final rejection.

SUMMARY OF CLAIMED SUBJECT MATTER

Independent claim 1 is directed to a processor comprising scheduling circuitry operative to schedule data blocks for transmission from a plurality of transmission elements. The scheduling circuitry is configurable for utilization of at least a first table and a second table in scheduling the data blocks for transmission. The processor further comprises memory circuitry associated with the scheduling circuitry and configurable to store at least a portion of at least one of the first and second tables.

The first table is configurable to include at least first and second lists of entries corresponding to transmission elements for which data blocks are to be scheduled in accordance with at least a first scheduling algorithm. The scheduler is operative to maintain a first table pointer identifying at least one of the first and second lists of the first table as having priority over the other of the first and second lists of the first table.

The second table is configurable to include a plurality of entries corresponding to transmission elements for which data blocks are to be scheduled in accordance with at least a

second scheduling algorithm different than the first scheduling algorithm. Association of a given one of the transmission elements with a particular one of the entries establishes a scheduling rate for that transmission element. The scheduler maintains a second table pointer identifying a current one of the second table entries as being eligible for transmission.

In an illustrative embodiment shown in FIG. 3, a processor (e.g., 102 in FIG. 2) comprises scheduling circuitry (e.g., scheduler 300 in FIG. 3) operative to schedule data blocks for transmission from a plurality of transmission elements (e.g., transmit queues 302 in FIG. 3); see, e.g., the specification at page 7, lines 11-19. The scheduling circuitry is configurable for utilization of at least a first table (e.g., FIFO lists 310) and a second table (e.g., dynamic calendar table 312) in scheduling the data blocks for transmission; see, e.g., the specification at page 7, lines 19-22. The processor further comprises memory circuitry (e.g., internal memory 104 in FIG. 1) associated with the scheduling circuitry and configurable to store at least a portion of at least one of the first and second tables; see, e.g., the specification at page 8, lines 13-17.

As described in the specification at page 9, line 26, to page 10, line 15, with respect to an illustrative embodiment shown in FIG. 4, the first table is configurable to include at least first and second lists (e.g., FIFO List 1, FIFO List 2, etc. in FIG. 4) of entries corresponding to transmission elements for which data blocks are to be scheduled in accordance with at least a first scheduling algorithm. The scheduler is operative to maintain a first table pointer (e.g., ActiveList Pointer in FIG. 4) identifying at least one of the first and second lists of the first table as having priority over the other of the first and second lists of the first table.

As described in the specification at page 11, line 14-24, with respect to an illustrative embodiment shown in FIG. 5, the second table is configurable to include a plurality of entries corresponding to transmission elements for which data blocks are to be scheduled in accordance with at least a second scheduling algorithm different than the first scheduling algorithm. Association of a given one of the transmission elements with a particular one of the entries establishes a scheduling rate for that transmission element. The scheduler maintains a second table pointer (e.g., CurrentPointer in FIG. 5) identifying a current one of the second table entries as being eligible for transmission.

Independent claim 14 is directed to a method for use in a processor. The method comprises storing at least a portion of at least one of a first table and a second table; and scheduling data blocks for transmission from a plurality of transmission elements, utilizing the first and second tables.

The first table is configurable to include at least first and second lists of entries corresponding to transmission elements for which data blocks are to be scheduled in accordance with at least a first scheduling algorithm. A first table pointer identifies at least one of the first and second lists of the first table as having priority over the other of the first and second lists of the first table.

The second table is configurable to include a plurality of entries corresponding to transmission elements for which data blocks are to be scheduled in accordance with at least a second scheduling algorithm different than the first scheduling algorithm. Association of a given one of the transmission elements with a particular one of the entries establishes a scheduling rate for that transmission element. A second table pointer identifies a current one of the second table entries as being eligible for transmission.

As described with respect to an illustrative embodiment in FIG. 3, a method for use in a processor (e.g., processor 102 in FIG. 2) includes storing (e.g., within internal memory 104 in FIG. 1) at least a portion of at least one of a first table (e.g., FIFO lists 310 in FIG. 3) and a second table (e.g., dynamic calendar table 312 in FIG. 3); see, e.g., the specification at page 8, lines 13-17. The method also includes scheduling data blocks for transmission from a plurality of transmission elements (e.g., transmit queues 302 in FIG. 3) utilizing the first and second tables; see, e.g., the specification at page 7, lines 11-22.

As described in the specification at page 9, line 26, to page 10, line 15, with respect to an illustrative embodiment shown in FIG. 4, the first table is configurable to include at least first and second lists (e.g., FIFO List 1, FIFO List 2, etc. in FIG. 4) of entries corresponding to transmission elements for which data blocks are to be scheduled in accordance with at least a first scheduling algorithm. A first table pointer (e.g., ActiveList Pointer in FIG. 4) identifies at least one of the first and second lists of the first table as having priority over the other of the first and second lists of the first table.

As described in the specification at page 11, line 14-24, with respect to an illustrative embodiment shown in FIG. 5, the second table is configurable to include a plurality of entries corresponding to transmission elements for which data blocks are to be scheduled in accordance with at least a second scheduling algorithm different than the first scheduling algorithm. Association of a given one of the transmission elements with a particular one of the entries establishes a scheduling rate for that transmission element. A second table pointer (e.g., CurrentPointer in FIG. 5) identifies a current one of the second table entries as being eligible for transmission.

Claim 20 is directed to an article of manufacture comprising a computer-readable medium for use in conjunction with a processor. The medium storing one or more software programs for use in scheduling data blocks for transmission from a plurality of transmission elements. The one or more programs when executed implement the step of scheduling data blocks for transmission from a plurality of transmission elements, utilizing first and second tables.

The first table is configurable to include at least first and second lists of entries corresponding to transmission elements for which data blocks are to be scheduled in accordance with at least a first scheduling algorithm. A first table pointer identifies at least one of the first and second lists of the first table as having priority over the other of the first and second lists of the first table.

The second table is configurable to include a plurality of entries corresponding to transmission elements for which data blocks are to be scheduled in accordance with at least a second scheduling algorithm different than the first scheduling algorithm. Association of a given one of the transmission elements with a particular one of the entries establishes a scheduling rate for that transmission element. A second table pointer identifies a current one of the second table entries as being eligible for transmission.

As described in the specification at, for example, page 7, lines 3-10, with respect to an illustrative embodiment shown in FIG. 2, an article of manufacture may comprise a computer-readable medium for use in conjunction with a processor (e.g., processor 102). As described in the specification at, for example, page 7, lines 11-22, with respect to an illustrative embodiment

shown in FIG. 3, the medium stores one or more software programs for use in scheduling data blocks for transmission from a plurality of transmission elements (e.g., transmit queues 302). The one or more programs, when executed, implement the step of scheduling data blocks for transmission from a plurality of transmission elements, utilizing first (e.g., FIFO lists 310) and second (e.g., dynamic calendar table 312) tables.

As described in the specification at page 9, line 26, to page 10, line 15, with respect to an illustrative embodiment shown in FIG. 4, the first table is configurable to include at least first and second lists (e.g., FIFO List 1, FIFO List 2, etc. in FIG. 4) of entries corresponding to transmission elements for which data blocks are to be scheduled in accordance with at least a first scheduling algorithm. A first table pointer (e.g., ActiveList Pointer in FIG. 4) identifies at least one of the first and second lists of the first table as having priority over the other of the first and second lists of the first table.

As described in the specification at page 11, line 14-24, with respect to an illustrative embodiment shown in FIG. 5, the second table is configurable to include a plurality of entries corresponding to transmission elements for which data blocks are to be scheduled in accordance with at least a second scheduling algorithm different than the first scheduling algorithm. Association of a given one of the transmission elements with a particular one of the entries establishes a scheduling rate for that transmission element. A second table pointer (e.g., CurrentPointer in FIG. 5) identifies a current one of the second table entries as being eligible for transmission.

Illustrative embodiments of the claimed invention provide a number of significant advantages over conventional arrangements. As discussed in the specification at, for example, page 2, lines 1-8, and page 6, lines 4-14, the present invention in the illustrative embodiment provides an efficient and flexible scheduler architecture capable of supporting multiple scheduling algorithms. The architecture is particularly advantageous the network processor context in that it provides sufficient flexibility to allow the implementation of new algorithms as they are defined without requiring redesign of the hardware or other elements of the network processor. Rather, a wide variety of scheduling algorithms may be implemented within a single

network processor, under software control, thereby avoiding the need for separate hardware architectures to support each of the desired scheduling algorithms.

GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

1. Claims 1-19 are rejected under 35 U.S.C. §103(a) as being unpatentable over U.S. Patent No. 7,120,153 (hereinafter "Anconetani").

2. Claim 20 is rejected under 35 U.S.C. §103(a) as being unpatentable over Anconetani in view of U.S. Patent No. 7,085,279 (hereinafter "Kumar").

ARGUMENT

1. Rejection of Claims 1-19 under §103(a) over Anconetani

Claims 1-10 and 14-19

Appellants submit that the Examiner has failed to establish a proper *prima facie* case of obviousness in the §103(a) rejection of claims 1-19 over Anconetani, in that the cited reference fails to teach or suggest all the limitations of these claims, and in that no cogent motivation has been identified for modifying the reference teachings to reach the claimed invention. Moreover, even if it is assumed for purposes of argument that a *prima facie* case has been established, there are teachings in the Anconetani reference that overcome any such *prima facie* case by directly teaching away from the claimed invention.

Independent claim 1 includes a limitation wherein the second table is configurable to include a plurality of entries corresponding to transmission elements for which data blocks are to be scheduled in accordance with at least a second scheduling algorithm different than the first scheduling algorithm, where association of a given one of the transmission elements with a particular one of the entries establishes a scheduling rate for that transmission element.

In an illustrative embodiment described in conjunction with FIGS. 3, 4 and 5 of the drawings, the first and second tables of claim 1 are shown, by way of example, as FIFO lists 310 and dynamic calendar table 312. A given queue or other transmission element having at least one data block to transmit can be scheduled either in one of the FIFO lists or in the dynamic calendar table. Thus, at a particular point in time, the given queue may be empty, that is, have no

data block to transmit, or may be scheduled via one of the FIFO lists or the dynamic calendar table. This advantageously allows multiple scheduling algorithms to be supported in an efficient and flexible manner. For example, the FIFO lists 310 may be used to implement a weighted fair queuing algorithm, while the dynamic calendar table 312 may be used to implement a constant bit rate or variable bit rate scheduling algorithm. See the specification at, for example, page 9, line 17, to page 12, line 13.

In the final Office Action at page 7, last paragraph, and in the Advisory Action at page 2, third paragraph, the Examiner argues, with reference to FIG. 11 of Anconetani, that context table 202 is configured to use a second scheduling algorithm, context logic 204, which is different from the first scheduling algorithm, calendar logic 208. Appellants respectfully disagree with the Examiner's characterization of Anconetani. Anconetani at column 14, lines 28-32, expressly indicates that the Calendar Schedule process of calendar logic 208 "is simply a response to a calendar schedule request from Context Logic 202," rather than being a different scheduling algorithm. In other words, calendar logic 208 uses the same scheduling algorithm as context logic 202. Accordingly, Anconetani fails to teach or suggest the limitations of claim 1 directed to at least a second scheduling algorithm different than the first scheduling algorithm.

Claim 1 recites a limitation wherein the second table is configurable to include a plurality of entries corresponding to transmission elements for which data blocks are to be scheduled. Here, the Examiner argues that the entries of the context table in Anconetani "correspond or relate to transmission elements for data blocks to be transmitted." Claim 1 also includes a limitation wherein "association of a given one of the transmission elements with a particular one of the entries establishes a scheduling rate for that transmission element." Appellants respectfully disagree and note that Anconetani at column 9, lines 57-60, clearly indicates that the context table stores context information for all connections assigned to cell scheduler 102, even connections which are idle (i.e., those for which no data blocks are to be scheduled). As such, Anconetani fails to teach or suggest that association of a given one of the transmission elements with a particular one of the entries establishes a scheduling rate for that transmission element, as recited in claim 1. Rather, context table 202 thus appears to provide what is generally referred to

in the art as traffic shaping information. See Anconetani at, for example, column 7, lines 33-44, and the present specification at, for example, page 9, lines 7-16.

Appellants further traverse the use of Official Notice with regard to the recited first and second pointers. The recited pointers are particular types of pointers, not simply pointers in general, and the use of Official Notice to attempt to meet these claim elements is believed to be inappropriate. For example, the recited first table pointer identifies one of the first and second lists of entries of the first table as having priority over the other of the first and second lists of entries. The recited second table pointer identifies a current one of the second table entries as being eligible for transmission. The Official Notice relates to the general use of pointers "to identify the packets ready for transmission" and thus fails to meet the particular recited first and second table pointers of claim 1. The reference relied upon in the final Office Action at page 8, last paragraph, for "the use of pointers in scheduling," likewise fails to teach or suggest the particular recited first and second table pointers of claim 1.

Accordingly, it is believed that the teachings of the Anconetani reference fail to meet the limitations of independent claim 1.

Furthermore, it is believed that insufficient objective evidence of motivation to modify Anconetani has been identified by the Examiner. The Examiner at page 3, third paragraph, of the final Office Action argues that one skilled in the art would be motivated to modify Anconetani to reach the particular limitations of claim 1 because "the use of pointers is known to provide smaller and faster algorithm execution in the memory of devices."

Appellants respectfully submit that the proffered statement fails to provide sufficient objective motivation and is instead a conclusory statement of the sort rejected by both the Federal Circuit and the U.S. Supreme Court. See KSR v. Teleflex, 127 S. Ct. 1727, 1741 (2007), quoting In re Kahn, 441 F. 3d 977, 988 (Fed. Cir. 2006) ("[R]ejections on obviousness grounds cannot be sustained by mere conclusory statements; instead, there must be some articulated reasoning with some rational underpinning to support the legal conclusion of obviousness."). As noted above, the specific pointer limitations of claim 1 are not met by Anconetani, and the proffered conclusory statement regarding the general use of pointers in memory fails to supplement this fundamental deficiency of Anconetani as applied to claim 1.

Independent claim 1 is therefore believed to be patentable over Anconetani.

Independent claim 19 is patentable for reasons similar to those outlined above with regard to claim 1.

Dependent claims 2-10 and 14-18 are patentable for at least the reasons identified above with regard to claim 1.

Claims 11 and 12

In addition to being patentable because of its dependency on independent claim 1, claim 11 is also believed to define separately-patentable subject matter. Specifically, claim 11 includes a limitation wherein a desired scheduling rate for a given one of the transmission elements is established by entering an identifier of that element into a particular one of the entries of the second table, the particular entry being determined as a function of the second table pointer and a designated scheduling interval.

In an illustrative embodiment described in the present specification at page 11, lines 1-24, with reference to FIG. 5, a queue identifier is placed in a slot denoted by Scheduled Slot = CurrentPointer + Interval. When a queue is scheduled in this manner, it is not eligible to transmit until the CurrentPointer reaches the slot that stores the identifier of that queue. The scheduling rate is thus determined by which of the table slots are assigned to that queue.

In formulating the rejection of claim 11 in the final Office Action at page 4, seventh paragraph, the Examiner contends that this limitation is met by page 9, lines 48-52, which the Examiner characterizes as disclosing "that the context table can hold a leaky bucket state for each VC that is evaluated periodically and that the context table also holds a separate TPCR [sic] for transmission rates."

Anconetani teaches an arrangement wherein the context table stores an entry for each connection in the cell scheduler and that each entry includes fields denoted as X, which stores a leaky bucket state for that connection, and T_{PCR}, which specifies the minimum period for that connection. The reciprocal of the T_{PCR} for a connection is the peak rate for that connection. See Anconetani at column 14, lines 15-22; see also Anconetani at column 9, lines 48-59; see

generally Anconetani at column 16, lines 30-59. It should be noted that neither T_{PCR} nor X is an identifier of a connection, but rather each represents a characteristic of a connection.

Thus, Anconetani teaches a technique wherein a peak rate for a connection is specified by numeric values, other than an identifier of a connection, within the context table entry for that connection. This technique fails to teach or even suggest the limitation of claim 11 in which the desired scheduling rate for a given one of the transmission elements is established by entering an identifier of that element into a particular one of the entries of the second table. Moreover, nowhere does Anconetani teach or even suggest the limitation wherein the particular entry is determined as a function of the second table pointer and a designated scheduling interval.

Claim 12 is patentable at least because of its dependency on claims 1 and 11.

Claim 13

In addition to being patentable because of its dependency on independent claim 1, claim 13 is also believed to define separately-patentable subject matter. Specifically, claim 13 includes a limitation wherein wherein each of the transmission elements at a given point in time may have a corresponding entry in the first table or in the second table, but not in both the first table and the second table.

In formulating the rejection of claim 13 in the final Office Action at page 5, first paragraph, the Examiner argues that "Anconetani further discloses that a data block moves through the scheduler and is not in a calendar table and a context table at the same time (Figure 14)." Appellants note that Anconetani at column 17, lines 19-21, describes FIG. 14 as illustrating successive operations on the same cell across multiple OC-12 cycles as a VCX propagates through ESI Scheduler 102. However, nowhere does FIG. 14 disclose the limitation of claim 13 wherein no transmission element has a corresponding entry in both the first table and the second table at the same time.

Indeed, Anconetani teaches directly away from this limitation by instead teaching arrangements wherein a connection may simultaneously have an entry in the context table for the connection itself and entries in the calendar table for cells to be transmitted for that connection. Indeed, the maximum number of entries which a connection may have in the calendar table is

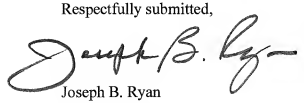
given by the Max parameter of that connection's entry in the context table. See, for example, column 9, lines 57-60; column 10, lines 44-46; and column 10, lines 63-66.

2. Rejection of Claim 20 under §103(a) over Anconetani and Kumar

Independent claim 20 contains limitations similar to those recited in claim 1, and is therefore believed patentable for reasons similar to those outlined above with regard to claim 1. The Kumar reference as applied to claim 20 fails to supplement the above-identified fundamental deficiencies of the Anconetani reference.

In view of the above, Appellants believe that claims 1-20 are in condition for allowance, and respectfully request the withdrawal of the § 103(a) rejections.

Respectfully submitted,



Date: June 9, 2008

Joseph B. Ryan
Attorney for Applicant(s)
Reg. No. 37,922
Ryan, Mason & Lewis, LLP
90 Forest Avenue
Locust Valley, NY 11560
(516) 759-7517

CLAIMS APPENDIX

1. A processor comprising:

scheduling circuitry operative to schedule data blocks for transmission from a plurality of transmission elements, the scheduling circuitry being configurable for utilization of at least a first table and a second table in scheduling the data blocks for transmission; and

memory circuitry associated with the scheduling circuitry and configurable to store at least a portion of at least one of the first and second tables;

the first table configurable to include at least first and second lists of entries corresponding to transmission elements for which data blocks are to be scheduled in accordance with at least a first scheduling algorithm, the scheduler being operative to maintain a first table pointer identifying at least one of the first and second lists of the first table as having priority over the other of the first and second lists of the first table;

the second table configurable to include a plurality of entries corresponding to transmission elements for which data blocks are to be scheduled in accordance with at least a second scheduling algorithm different than the first scheduling algorithm, association of a given one of the transmission elements with a particular one of the entries establishing a scheduling rate for that transmission element, the scheduler maintaining a second table pointer identifying a current one of the second table entries as being eligible for transmission.

2. The processor of claim 1 wherein the first table comprises a plurality of first-in first-out lists, the plurality of lists including the first and second lists and at least one additional list.

3. The processor of claim 1 wherein the first table pointer comprises an active list pointer identifying one of the first and second lists as an active list.

4. The processor of claim 1 wherein in scheduling data blocks for transmission utilizing the first table, the scheduling circuitry identifies a first entry in a first non-empty one of the lists starting from a list identified by the first table pointer, and schedules for transmission a data block from the corresponding transmission element.

5. The processor of claim 1 wherein one of the first and second lists of the first table comprises an active list and the other of the first and second lists of the first table comprises a pending list, wherein the active list comprises a list of one or more of the transmission elements which have not exceeded corresponding bandwidth allocations, and the pending list comprises a list of one or more of the transmission elements which have exceeded corresponding bandwidth allocations, the bandwidth allocations being determined over a programmable time interval based on relative rate partitions between the transmission elements.

6. The processor of claim 5 wherein the first table pointer initially points to the active list, and when the active list becomes empty, the pointer is updated to point to the pending list, the pending list is designated as a new active list, and the previous active list is designated as a new pending list.

7. The processor of claim 5 wherein a given transmission element which has exceeded its corresponding bandwidth allocation is moved from the first table to the second table in order to provide an adjustment in its scheduling rate.

8. The processor of claim 1 wherein the second table includes a plurality of slots, with each slot capable of storing the identifier of one of the transmission elements.

9. The processor of claim 1 wherein the second table pointer comprises a current pointer which is incremented in accordance with a time base of the scheduling circuitry.

10. The processor of claim 1 wherein the second table is configured such that each of the plurality of transmission elements can be dynamically assigned a transmission rate.

11. The processor of claim 1 wherein a desired scheduling rate for a given one of the transmission elements is established by entering an identifier of that element into a particular one of the entries of the second table, the particular entry being determined as a function of the second table pointer and a designated scheduling interval.

12. The processor of claim 11 wherein the designated scheduling interval is dynamically alterable under software control.

13. The processor of claim 1 wherein each of the transmission elements at a given point in time may have a corresponding entry in the first table or in the second table, but not in both the first table and the second table.

14. The processor of claim 1 wherein the first scheduling algorithm comprises a weighted fair queuing scheduling algorithm, the weighted fair queuing algorithm being configurable to implement at least one of round robin scheduling, strict priority scheduling, weighted round robin scheduling, smooth weighted round robin scheduling and smooth deficit weighted round robin scheduling.

15. The processor of claim 1 wherein the second scheduling algorithm comprises at least one of a constant bit rate scheduling algorithm and a variable bit rate scheduling algorithm.

16. The processor of claim 1 wherein the memory circuitry comprises at least one of internal memory and external memory of the processor.

17. The processor of claim 1 wherein one or more of the data blocks comprise data packets.

18. The processor of claim 1 wherein the processor comprises a network processor integrated circuit configured to provide an interface for data block transfer between a network and a switch fabric.

19. A method for use in a processor, the method comprising:

storing at least a portion of at least one of a first table and a second table; and

scheduling data blocks for transmission from a plurality of transmission elements, utilizing the first and second tables;

the first table configurable to include at least first and second lists of entries corresponding to transmission elements for which data blocks are to be scheduled in accordance with at least a first scheduling algorithm;

a first table pointer identifying at least one of the first and second lists of the first table as having priority over the other of the first and second lists of the first table;

the second table configurable to include a plurality of entries corresponding to transmission elements for which data blocks are to be scheduled in accordance with at least a second scheduling algorithm different than the first scheduling algorithm, association of a given one of the transmission elements with a particular one of the entries establishing a scheduling rate for that transmission element;

a second table pointer identifying a current one of the second table entries as being eligible for transmission.

20. An article of manufacture comprising a computer-readable medium for use in conjunction with a processor, the medium storing one or more software programs for use in scheduling data blocks for transmission from a plurality of transmission elements, the one or more programs when executed implementing the step of:

scheduling data blocks for transmission from a plurality of transmission elements, utilizing first and second tables;

the first table configurable to include at least first and second lists of entries corresponding to transmission elements for which data blocks are to be scheduled in accordance with at least a first scheduling algorithm;

a first table pointer identifying at least one of the first and second lists of the first table as having priority over the other of the first and second lists of the first table;

the second table configurable to include a plurality of entries corresponding to transmission elements for which data blocks are to be scheduled in accordance with at least a second scheduling algorithm different than the first scheduling algorithm, association of a given one of the transmission elements with a particular one of the entries establishing a scheduling rate for that transmission element;

a second table pointer identifying a current one of the second table entries as being eligible for transmission.

EVIDENCE APPENDIX

None

RELATED PROCEEDINGS APPENDIX

No decision has been rendered in the related proceeding identified above.